Remarks

The present amendment responds to the Official Action dated June 23, 2008. The Official Action objected to the title. Claims 9 and 18-26 were rejected under 35 U.S.C. § 103(a) based on Sheaffer U.S. Patent No. 6,957,321 (Sheaffer) in view of Miller et al. U.S. Patent No. 6,847,365 (Miller). Claims 10, 11, 13, and 15 were rejected under 35 U.S.C. § 103(a) based on Sheaffer in view of Miller and further in view of Moller et al. U.S. Patent No. 6,826,522 (Moller). Claims 12, 14, and 16 were rejected under 35 U.S.C. § 103(a) based on Sheaffer in view of Miller and in view of Moller and further in view of Tremblay U.S. Patent No. 6,341,348 (Tremblay). Claim 17 was rejected under 35 U.S.C. § 103(a) based on Sheaffer in view of Miller and further in view of Pechanek U.S. Patent No. 6,173,389 (Pechanek '389).
These grounds of rejection are addressed below. Claims 1-8 were previously canceled without prejudice. New claims 27-31 have been added. Claims 9, 11, 13, 15, 17, 18, 20-23, and 25 have been amended to be more clear and distinct. Claims 9-31 are presently pending.

Amendments to the Specification:

The title has been changed to respond to the objection.

A typo was recognized at page 21, lines 19 and 20. The sentence "The VIMBasket 1210 consists of either four slots 1243 or six slots 1244 in Fig. 12 instead of the fixed five slot arrangement in Fig. 5" has been corrected by amendment to read "The VIMBasket 1210 consists of either four slots ~~1243~~ 1244 or six slots ~~1244~~ 1243 in Fig. 12 instead of the fixed five slot

arrangement in Fig. 5". The amended text corresponds to the numbering shown in Fig. 12B and is also supported by the text at page 22, lines 14 and 15.

The Art Rejections

As addressed in greater detail below, Sheaffer, Miller, Moller, Tremblay, and Pechanek '389 do not support the Official Action's reading of them and the rejections based thereupon should be reconsidered and withdrawn. Further, the Applicants do not acquiesce in the analysis of Sheaffer, Miller, Moller, Tremblay, and Pechanek '389 made by the Official Action and respectfully traverse the Official Action's analysis underlying its rejections. More particularly, detailed analyses of the failings of Sheaffer, Moller, Tremblay, and Pechanek '389 have been previously provided in the Appeal Brief filed on April 28, 2008. These grounds are resubmitted here, but not repeated in the interest of brevity.

Claims 9 and 18-26 were rejected under 35 U.S.C. § 103(a) based on Sheaffer in view of Miller.

Regarding claim 9, the Official Action correctly admits "the fact that Sheaffer has characterized" his instructions as "two separate instructions, rather than an extended instruction". Official Action, Response to Arguments at page 12, lines 10-12. More particularly, Sheaffer's two separate instructions require that two separate opcodes be decoded to identify the functionality specified by the combination of the two separate opcodes. For example, "one opcode may be designated as a prefix" and the prefix is paired with a subsequent instruction opcode to change "the meaning of at least one subsequent instruction opcode", as described at

Shaeffer, col. 1, lines 33-38. Claim 9 has been amended to more clearly distinguish Sheaffer. Sheaffer's two separate instructions each with its own opcode are not an "expanded width instruction has a single opcode of contiguous bits" as claimed in amended claim 9. Emphasis added. It is noted that having a "single opcode of contiguous bits" in an expanded width instruction is advantageous in creating a super set of standard width instructions. For example, a standard width multiply accumulate unit (MAU) instruction formatted in a 32-bit encoding 600 having a standard width MAU opcode is shown in Fig. 6A of the present invention. An expanded width instruction formatted in a 40-bit encoding 610 having an expanded width MAU opcode is shown in Fig. 6B. As shown, for example, in Fig. 6C, a mapping 620 illustrates the 32-bit encoding 600 mapped to the 40-bit encoding 610. In the mapping 620, the "extended instruction fields are encoded to a '0' value preserving the present opcodes, register file addressing ranges and existing data types" as described at page 14, line 21 – page 15, line 9. The standard width instruction is preserved in an expanded width instruction slot while still allowing the expanded width instruction having "a single opcode of contiguous bits" to be loaded in the same expanded width slot, as claimed in claim 9. Sheaffer does not teach and does not make obvious the elements of claim 9.

In the Official Action, at page 3, lines 4 and 5, the Official Action suggests that Sheaffer addresses a VLIW memory configured for loading said at least one expanded format slot with an expanded instruction at Sheaffer, col. 7, lines 1-3. However, at the cited text, Sheaffer describes a "stream of instructions as stored in sequentially fetched memory addresses" and does not address a VLIW memory. In this regard, the Official Action, at page 3, lines 6 and 7 correctly

admits that "Sheaffer fails to disclose that VLIW memory and a standard size VLIW instruction". The Official Action, Response to Arguments at page 12, line 22 – page 13, line 1 also correctly admits that Sheaffer "fails to disclose a 'VLIW memory having a plurality of instruction slots for storing VLIW instruction words". Miller at col. 13, lines 1-4 is cited to purportedly address these discrepancies. At Miller, col. 13, lines 1-4, Miller indicates instructions making up a VLIW may be of variable length. The Official Action then suggests the combination of Sheaffer's "normal instruction and a NOP" instruction in a "VLIW instruction for expanded operand capabilities". Official Action, at page 3, lines 15-17. The Official Action provides no further description of how Miller may be combined with Sheaffer. In particular, Miller is silent regarding the format of instructions and instruction opcodes that are included in his VLIW. Miller is also silent regarding standard width instructions and an expanded width instruction. Further, as noted above, Sheaffer's normal instruction and a NOP instruction, each with its own opcode, are not an expanded width instruction having a "single opcode of contiguous bits". Thus, Miller provides no basis for combining or modifying Sheaffer and does not resolve the admitted deficiencies of Sheaffer.

Also, Miller is silent regarding loading instructions in the VLIW memory and accessing an expanded width instruction from a local data memory separate from the VLIW memory as claimed in claim 9. Miller's data memory 122 shown in Fig. 4 does not appear to store instructions or store VLIWs. Miller's data memory 122 does not appear to be connected to his instruction memory 120. In particular, Miller utilizes a "parallel bus architecture having an instruction bus 102 and a data bus 104" where the instruction bus 102 "transmits instructions"

and the data bus 104 "transmits data", as described at Miller, col. 10, lines 19-23. Miller's "instructions and data do not travel over the same bus". Miller, col. 10, lines 24-26. Thus, an expanded width instruction having a "single opcode of contiguous bits" can not be accessed from the data memory 122 and loaded in an expanded width instruction slot in VLIW memory as claimed in claim 9. The combination of Sheaffer and Miller does not teach and does not make obvious the "VLIW memory configured for loading said at least one expanded width instruction slot with the expanded width instruction accessed from a local data memory separate from the VLIW memory" as claimed in claim 9.

Regarding claim 18, the Official Action suggests that Sheaffer combined with Miller discloses an instruction memory holding a plurality of instructions of a first bit width, the plurality of instructions having at least one execute VLIW instruction citing Miller col. 13, lines 1-4 for support. At the cited text, Miller indicates instructions making up a VLIW may be of variable length. This reference to Miller's VLIW instructions does not address an "instruction memory holding a plurality of instructions of a first bit width, the plurality of instructions having at least one execute VLIW instruction that specifies a VLIW address for fetching a VLIW for decoding and execution" as claimed in claim 18. If anything, Miller's VLIW instructions are stored in his instruction memory 120, "in the form of very long instruction word (VLIW) instructions" as described at Miller, col. 10, lines 38-41 with reference to Fig. 4. Miller shows another memory, data memory 122, in Fig. 4, but Miller's data memory 122 does not appear to store instructions or to store VLIWs. Miller does not teach and does not make obvious any other memory which may be considered "an instruction memory holding a plurality of instructions of

the first bit width" as claimed in claim 18. Also, Miller does not teach and does not make obvious an "execute VLIW instruction that specifies a VLIW address for fetching a VLIW for decoding and execution" as claimed in claim 18.

The Official Action further suggests that the combination of Sheaffer and Miller discloses a very long instruction memory having instruction slots for storing instructions of a second bit width different from the first bit width and wherein the very long instruction memory holds VLIWs at addressable locations that may be fetched as a result of executing the at least one execute VLIW instruction citing Sheaffer, col. 1, lines 33-50 or col. 2, line 65 to col. 3, line 7 and Miller, col. 13, lines 1-4 for support. Sheaffer at col. 1, lines 33-50 describes an "opcode may be designated as a prefix" and the "prefix changes the meaning of at least one subsequent instruction opcode". Thus, Sheaffer requires the use of two opcodes that are stored in a memory prior to decoding the two instructions. Sheaffer at col. 2, line 65 to col. 3, line 7 describes the use of a NOP instruction and another instruction, both of which are identified by their associated and separate opcodes. Miller, at col. 13, lines 1-4 merely discloses that instructions making up a VLIW may be of variable length which, as noted above, does not resolve the deficiencies of Sheaffer. These three citations, separately or in combination, do not meet the language of claim 18. In particular, the combination of Sheaffer and Miller does not teach and does not make obvious an "instruction of a second bit width having a single opcode of contiguous bits, and wherein ... the VLIW having the instruction of the second bit width being fetched from the very long instruction memory at the VLIW address as a result of executing the at least one execute VLIW instruction" as claimed in claim 18. Emphasis added.

As noted above, Miller does not teach and does not make obvious any memory which may be considered "an instruction memory holding a plurality of instructions of the first bit width" as claimed in claim 18. If in a hypothetical combination of Sheaffer and Miller, Sheaffer provides the "instruction memory holding a plurality of instructions of the first bit width" and Miller provides the "very long instruction memory ... for storing an instruction of the second bit width having a single opcode of contiguous bits" as claimed in claim 18, then the hypothetical combination could only operate the two memories separately and not function as claimed in claim 18. No coupling between the two separate memories is suggested or addressed in any form in either Sheaffer or in Miller.

Further, even if the two separate memories were hypothetically combined as a single combined memory, the single combined memory would not meet the requirements of the claim 18 as having two memories, "an instruction memory" and a "very long instruction memory". Also, Miller does not teach and does not make obvious an "execute VLIW instruction that specifies a VLIW address for fetching a VLIW for decoding and execution" that is one of the "plurality of instructions of the first bit width" as claimed in claim 18. Further, Miller does not teach and does not make obvious "the VLIW having the instruction of the second bit width being fetched from the very long instruction memory at the VLIW address as a result of executing the at least one execute VLIW instruction" as claimed in claim 18.

To sum up, the combination of Sheaffer and Miller does not teach and does not make obvious claim 18.

In rejecting claim 20, the Official Action relies upon the same sections of Sheaffer and Miller that are addressed above with regard to related elements of claim 18. As discussed in detail above, Sheaffer and Miller in combination do not teach and do not make obvious amended claim 20.

The Official Action additionally suggests that "on paragraph 53 of Applicant's specification, the instructions designated as 'compressed instructions' appear to be essentially the same as the extended instructions, just through a different view point. As claimed (in light of the Applicant's specification), it appears reasonable to consider the non-extended instruction a compressed instruction." Paragraph 53 of the Applicants' specification corresponds to the paragraph beginning at page 13, line 7 and addresses instruction formats smaller than 32-bits that "can be stored in the VLIW memory slots, as might be useful in a compressed instruction system". By way of example, a program storage holding standard width instructions is shown in Applicant's exemplary ManArray 2x2 iVLIW single instruction multiple data (SIMD) processor 100 of Fig. 3 as the 32-bit instruction memory 105. In addition, the short instruction word (SIW) "I-Fetch controller 103 dispatches 32-bit SIWs to the other PEs in the system by means of the C=32-bit instruction bus 102" as described at page 9, lines 2 and 3. Further, Applicants' Fig. 9A and text at page 18, lines 14-19 addresses a dual instruction format holding two 15-bit instructions 914 and 916 each of a compressed format smaller than 32-bits and a single 32-bit instruction encoding 922 of a standard width. As claimed in claim 20, the compressed instruction has "a narrower instruction format with respect to an instruction format of a standard

width required by program instructions stored in a program storage". Thus, the Official Action's suggestion (in light of the Applicant's specification) is not correct.

The combination of Sheaffer and Miller does not teach and does not make obvious "a plurality of instruction slots for storing instruction words forming a VLIW, at least one of said plurality of instruction slots having a compressed format for storing a compressed instruction having a narrower instruction format with respect to an instruction format of a standard width required by program instructions stored in a program storage, wherein the VLIW resides at an addressable location in a VLIW memory separate from the program storage, the VLIW memory holding VLIWs that may be fetched for execution" as claimed in claim 20. Emphasis added.

Regarding claim 21, the Official Action suggests that the term "indirect" used in claim 21 "appears to be referring to an 'indirect execution mechanism' in Applicant's specification. This mechanism is anticipated by the additional NOP operands that are indirectly used in execution". Official Action, page 5, lines 11-13. Again, the Official Action's suggestion is not correct. Sheaffer's NOP operands specify addresses of data operands to be used when executing an instruction. Sheaffer's NOP operands do not anticipate "the expanded VLIW fetched from the indirect VLIW memory at a VLIW address generated in response to an execute VLIW short instruction word dispatched from the memory" as claimed in amended claim 21.

As discussed above with regard to claim 9, Sheaffer in combination with Miller does not teach and does not make obvious an "expanded instruction format comprise a single opcode of contiguous bits" as claimed in claim 21.

Regarding claim 17, the Official Action suggests that we are to "see below" for discussion of claim 17's "instruction bit organizer for receiving instructions as data and based on the load mask bit field organizing the bits from the data encoded instructions into proper format for loading into the specified instruction slots in the VIMB". However, the Official Action makes no further reference to the "instruction bit organizer" of claim 17.

The Official Action correctly admits that the combination of Sheaffer and Miller fails to disclose a load indirect instruction or a mask bit field specifying which slots will be loaded in a VLIW having at least one instruction slot that is an expanded instruction slot, the VLIW accessible to be loaded at an addressable location into the VIMB. Pechanek '389, at col. 3, lines 24-33 is depended upon to address the admitted deficiencies. However, Pechanek '389 at col. 3, lines 13-33 addresses an execute VLIW instruction that specifies indirect selection of a VLIW having multiple slot instructions to be read out of a VLIW memory for execution of mask enabled slot instructions. The function of executing a VLIW caused by Pechanek '389's execute VLIW instruction is completely different than the function of loading a VLIW caused by the claimed load indirect VLIW (LV) instruction. Emphasis added. Pechanek '389 does not describe a "load indirect VLIW (LV) instruction comprising data address information and a load mask bit field specifying which instructions are to be loaded in a VLIW" as claimed in claim 17. Thus, Pechanek '389 does not resolve the admitted deficiencies of Sheaffer and Miller. The combination of Sheaffer, Miller, and Pechanek '389 does not teach and does not make obvious an instruction bit organizer which "receives instructions as data from the local data memory according to the data address information and based on the load mask bit field organizes the bits

from the data encoded instructions into proper format to be loaded into the specified instruction slots in the VIMB in response to execution of the LV instruction" as claimed in claim 17.

Conclusion

All of the presently pending claims, as amended, appearing to define over the applied references, withdrawal of the present rejection and prompt allowance are requested.

Respectfully submitted,

Peter H. Priest
Reg. No. 30,210
Priest & Goldstein, PLLC
5015 Southpark Drive, Suite 230
Durham, NC 27713-7736
(919) 806-1600